COCOMO (and Agile)

Question: is COCOMO valid for AGILE?

- No!!
- Agile is totally different
- COCOMO and Function Point analysis based on construction / build characteristics of the system that has to be developed
- Story points are a ***relative measurement***
  - Created by the team itself
  - Not related to objective criteria
- COCOMO is based on a design created beforehand
- AGILE has short **spurts of design** – based on short feedback cycles and
- AGILE benefits from new insights/creativity

COCOMO

- COCOMO ‘may create’ a feeling of ‘yes I can accurately estimate…’
  - This is not true
  - COCOMO does not adequately account for the level of uncertainty in a project
  - No project estimate can be highly accurate
  - Requirements change and vary
  - Uncertainty always there and needs to be communicated
- COCOMO is a bit ‘old fashioned’
- Some principles are useful but COCOMO is not used widely

AGILE PM

- **Product owner/manager** and **SCRUM Master**
- Work with a large amount of uncertainty
- Difficult to estimate
- Estimation easier after the first SPRINT
- 2 things to estimate:
  - i) STORY POINTS and
  - ii) VELOCITY

The Agile: Scrum Framework at a glance

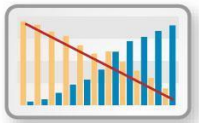
For all, By-Rahul Chaitanya

Inputs from Executives, Team, Stakeholders, Customers, Users

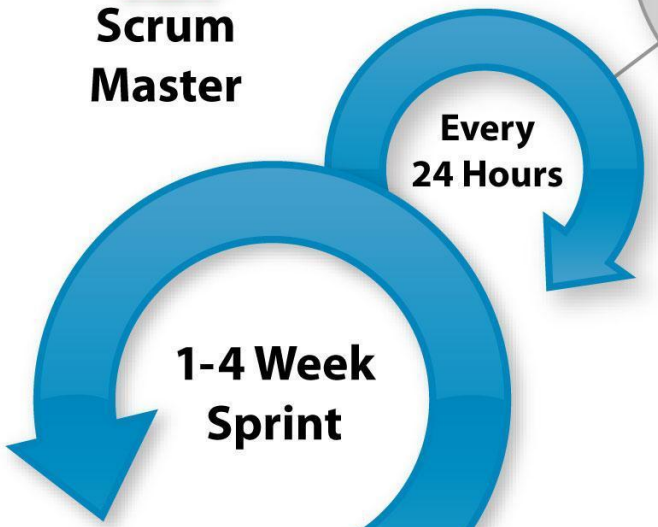
Product Owner

The Team

Scrum Master



Burndown/up Charts



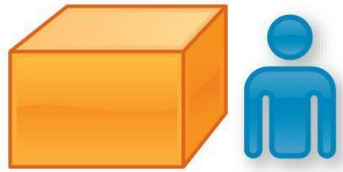
Sprint Review

Product Backlog

Sprint Planning Meeting

Sprint Backlog

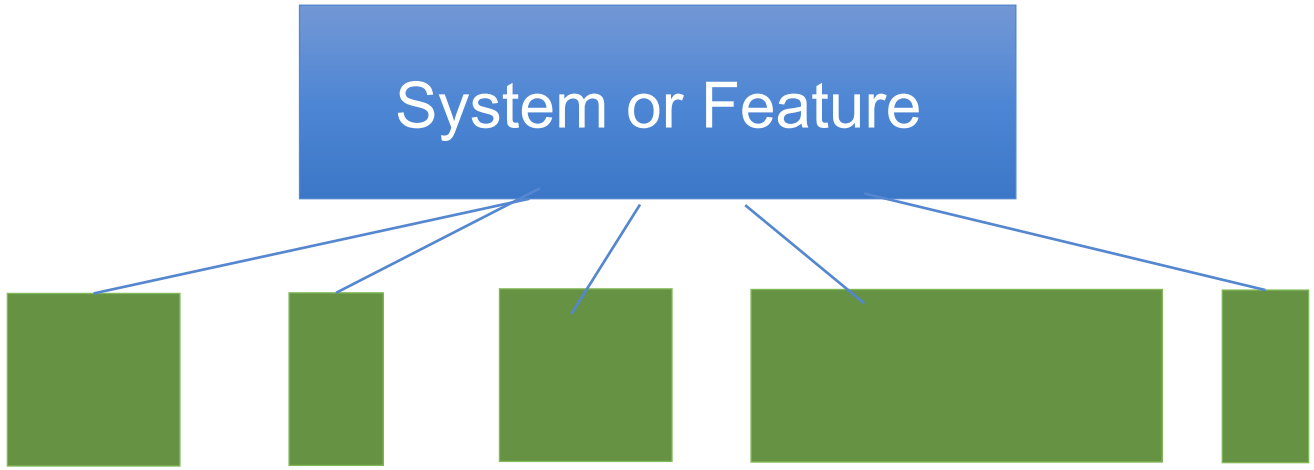
Sprint end date and team deliverable do not change



Finished Work

Sprint Retrospective

ESTIMATION IN AGILE



1. DIVIDE INTO STORIES

2. ESTIMATE NUMBER OF STORY POINTS PER STORY, BASE ESTIMATE ON NO OF STORY POINTS FROM PREVIOUS STORIES

3. USE TEAM VELOCITY FROM PRIOR STORIES TO ESTIMATE DELIVERY TIME OF PROJECT

4. DURING STORY DEVELOPMENT, MEASURE THE ACTUAL VELOCITY TAKEN BY THE TEAM

5. USE THIS VELOCITY, RE-ESTIMATE THE TIME IT WILL TAKE TO DELIVER THE PROJECT

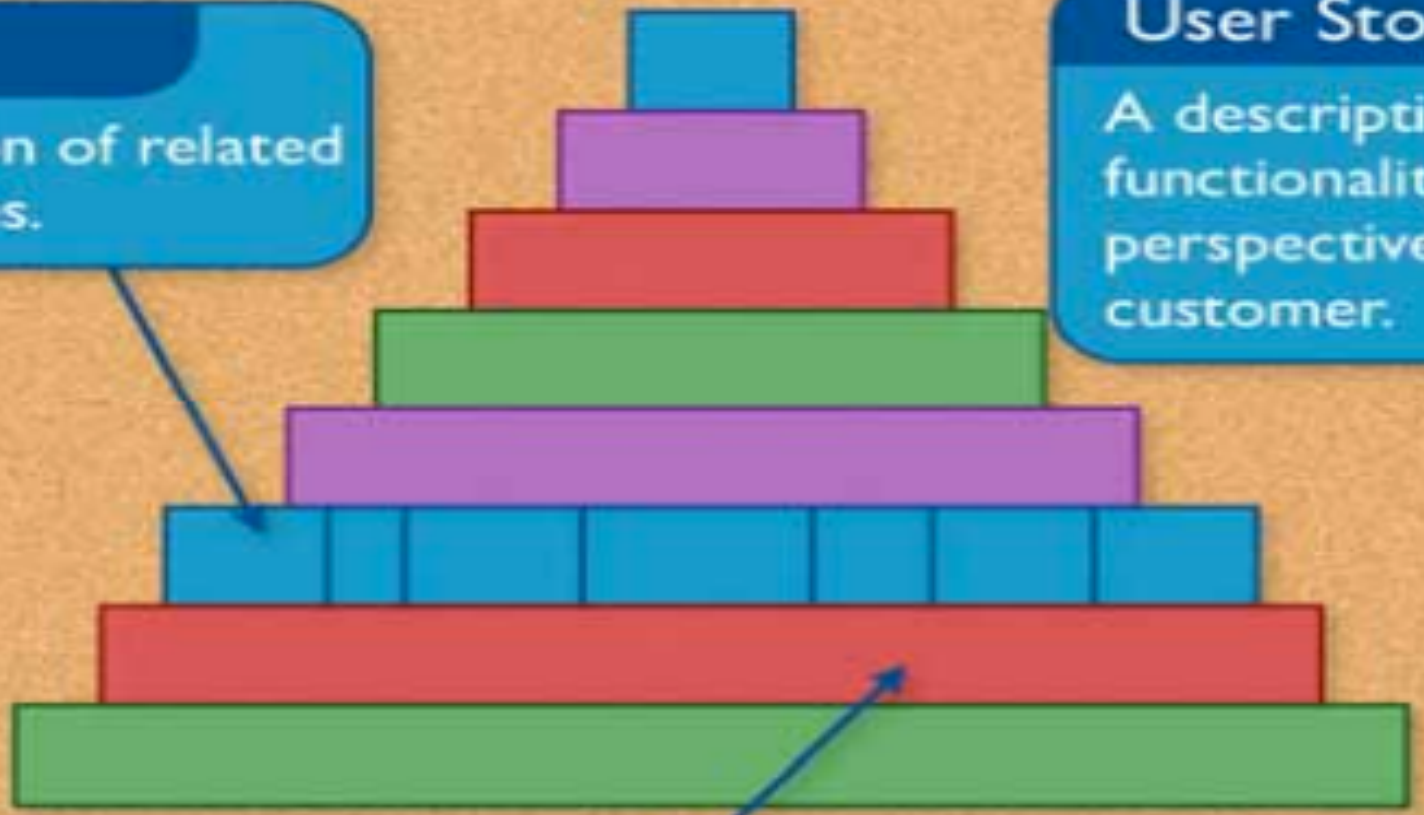
Stories, themes and epics

Theme

A collection of related user stories.

User Story

A description of desired functionality told from the perspective of the user or customer.



Epic

A large user story.

The product backlog

- The requirements
- A list of all desired work on the project
- Ideally expressed such that each item has value to the users or customers of the product
- Prioritized by the *product owner*
- Reprioritized at the start of each iteration

Sample user stories

As an account holder, I want to check my savings account balance.

As an account holder, I am required to authenticate myself before using the system.

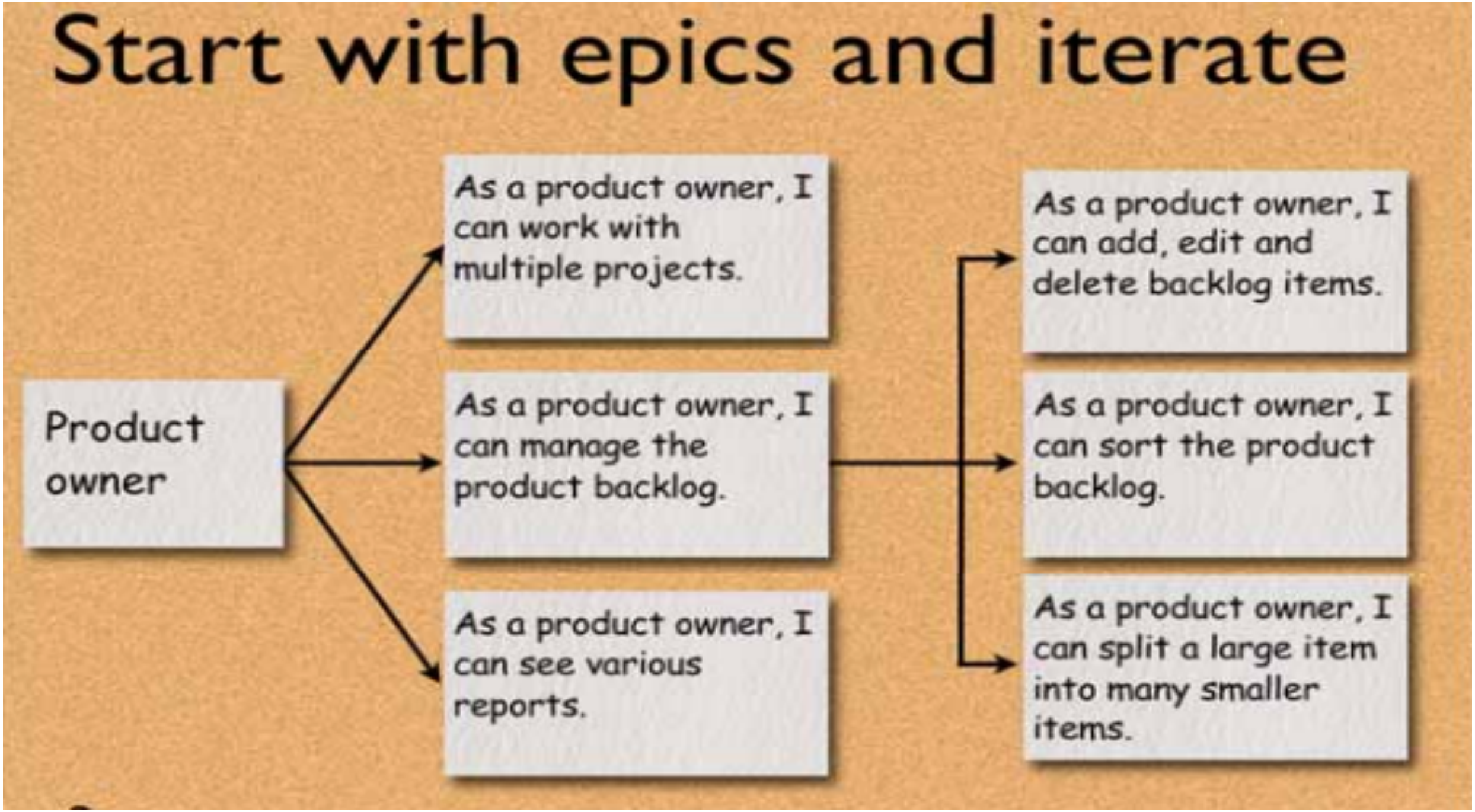
As the primary account holder, I can grant access to additional users so that they can see transactions.

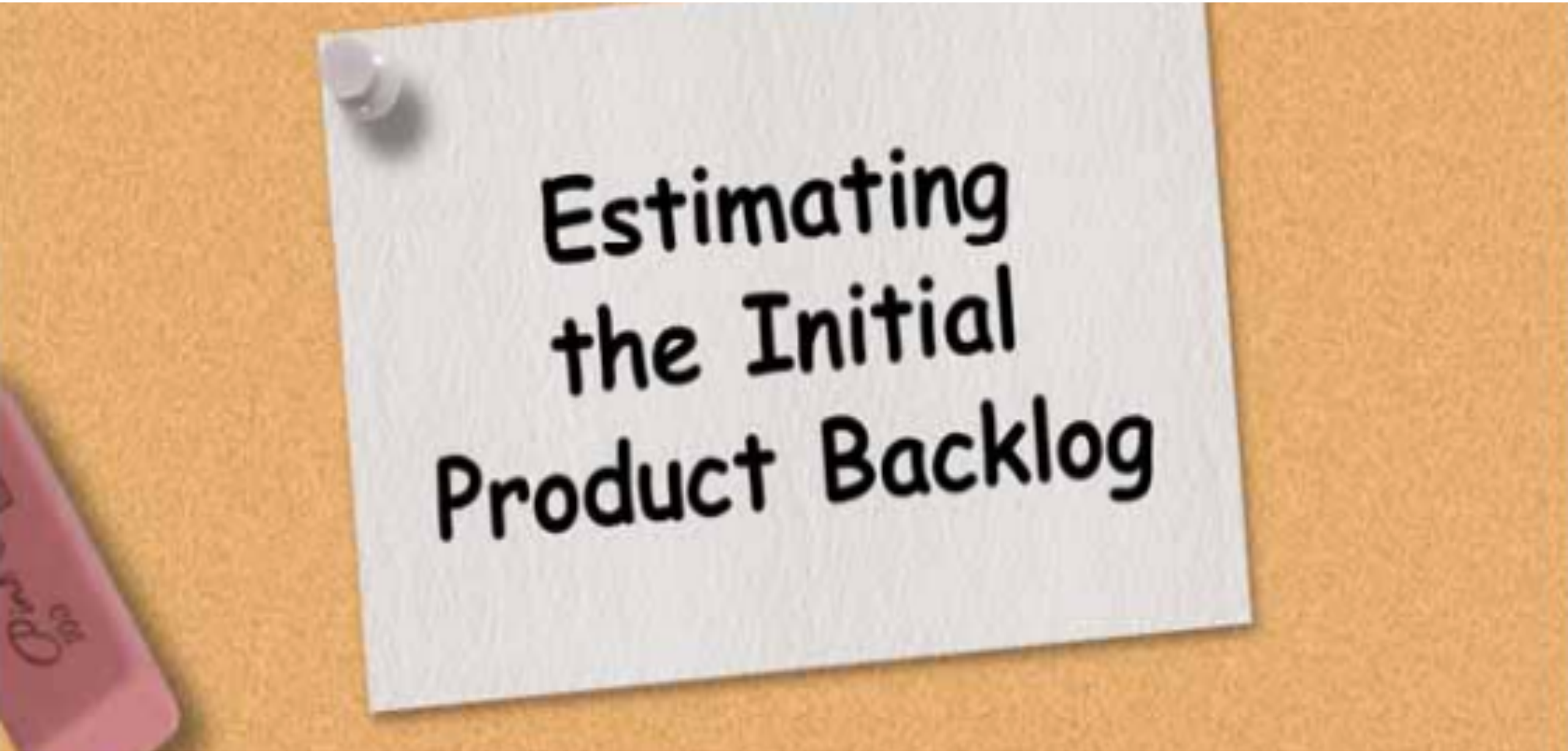
Non-functional user stories

As one of 10,000 concurrent users, I would like the system to perform adequately.

As a first-time user, I can complete common operations without using the help system.

Start with epics and iterate




Estimating
the Initial
Product Backlog

How long will it take...

- ...to read the latest Harry Potter book?
- ...to drive to Minneapolis?

Estimation of Agile

- Estimation problems b/w teams/organistions
- Team would calc different story points and would not agree - estimates are *relative to previous stories*
- **Guidelines to estimate story points:**
  - Estimation by analogy
  - Decompose a story
  - Use the right units
  - Doer does the estimation
  - Use group-based estimates

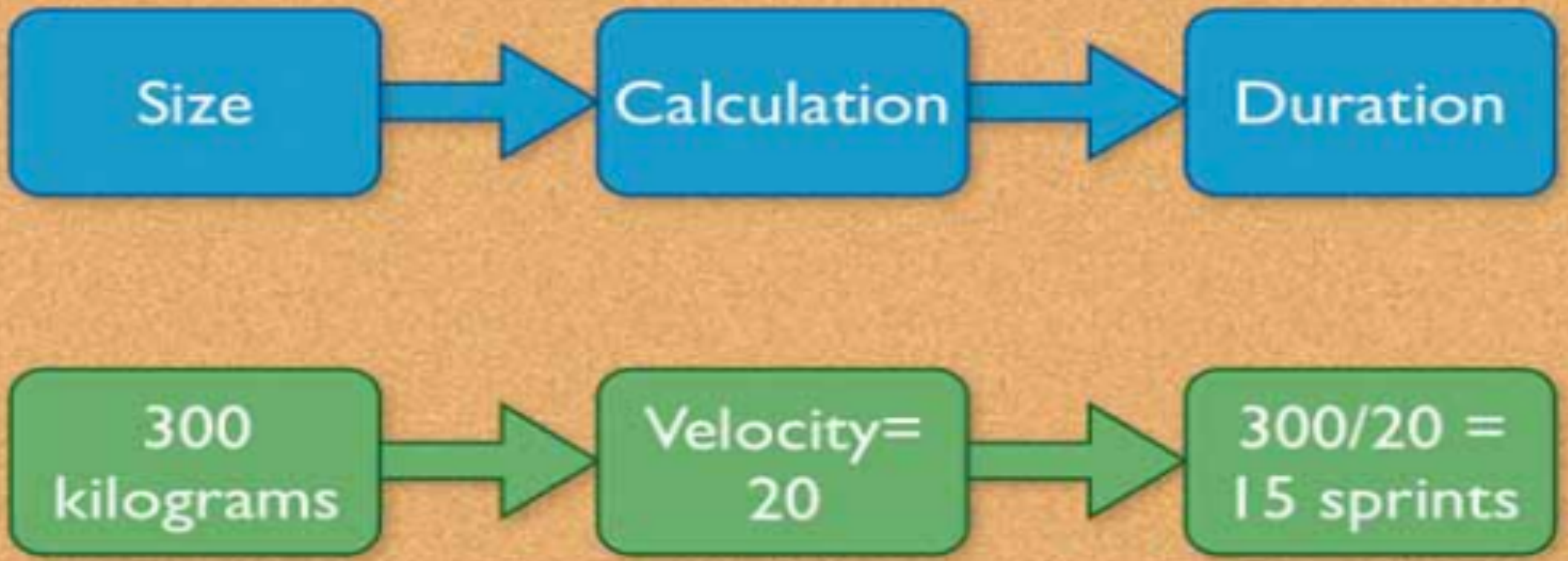
VELOCITY

- Team velocity is measured as the # of story points completed over a period

$$V = \frac{SP}{T_i},$$

- With SP the no of story points completed T_i the time period over which they were completed
- So e.g. a team complete 12 story points in one week VELOCITY = 12 sp's per week

Estimate size; derive duration



CALCUTING VELOCITY – 2 ways

1) Use historical data

– look @ team velocity (or similar team) over previous projects, then calc ave velocity of that team and use in estimates

2) Use data from previous iterations

- Wait until first iteration done, then calc velocity of team over that iteration and use this to estimate velocity of remainder of the projects

- UPDATE VELOCITY AFTER EACH ITERATION

ESTIMATE DELIVERY TIME

SUM the number of story points for ALL stories, and divide by velocity

$$T = \frac{\sum_{i=1}^{n} SP_i}{V},$$

USER STORY	STORY POINTS
Story 1	4 points
Story 2	8 points
Story 3	16 points
Story 4	16 points
Story 5	8 points
TOTAL	

ESTIMATE DELIVERY TIME

Team of 5 developing has an ave velocity of 6 user stories per fortnight. Thus initial est. for delivery is:

$$T = \frac{\sum_{i=1}^{n} SP_i}{V},$$

$$\begin{aligned} T_a &= \frac{\sum_{i=1}^{n} SP_i}{V} \\ &= \frac{52 \text{ points}}{6 \text{ points/fortnight}} \\ &= 8.66^* \text{ fortnights} \\ &\approx 17 \text{ weeks.} \end{aligned}$$

Re- calculate new DELIVERY TIME

Team takes 6 wks to complete the first two iterations. The first two stories together consists of 12 story points – V = 4 points per fortnight: Recalculate delivery time -see ch 7 p 96

$$T = \frac{\sum_{i=1}^{n} SP_i}{V},$$

$$\begin{aligned} T_b &= \frac{\sum_{i=1}^{n} SP_i}{V} \\ &= \frac{40 \text{ points}}{4 \text{ points/fortnight}} \\ &= 10 \text{ fortnights} \\ &= 20 \text{ weeks.} \end{aligned}$$

Ref’s

- Thank you MountainGoatSoftware website!